MNIST Classification Using Machine Learning Packages

Jiasui Ding & Brad Fanella

Testing Accuracy

|  |  |  |
| --- | --- | --- |
| **Package** | **Model** | **Accuracy** |
| sklearn | Random Forest Classifier | 0.9665 |
| sklearn | Support Vector Machine Classifier | 0.8811 |
| keras | 3-Layered MLP With All Hidden Layers Having a Size Of 256. | 0.9641 |
| keras | 2-Layered MLP With All Hidden Layers Having a Size Of 512. | 0.9604 |
| keras | 3-Layered MLP With the Hidden Layers’ Sizes Being 128, 256, and 512. | 0.9567 |
| keras | 3-Layered MLP With the Hidden Layers’ Sizes Being 512, 256, and 128. | 0.9570 |

# Part One: MNIST Classification Using Sci-Kit Learn

## Random Forest Classifier

Hyper-Parameters: bootstrap=True, class\_weight=None, criterion='gini',

max\_depth=None, max\_features=100, max\_leaf\_nodes=None,

min\_impurity\_decrease=0.0, min\_impurity\_split=None,

min\_samples\_leaf=1, min\_samples\_split=2,

min\_weight\_fraction\_leaf=0.0, n\_estimators=30, n\_jobs=None,

oob\_score=False, random\_state=None, verbose=0,

warm\_start=False

Note: The grid search chose the maximum value of 30 from parameter n\_estimators and the minimum value of 100 from parameter max\_features. We could try searching with a larger number of n\_estimators and/or a smaller number of max\_features to improve the score.

Support Vector Machine Classifier

Hyper-Parameters: C=1.0, cache\_size=200, class\_weight=None, coef0=0.0,

decision\_function\_shape='ovr', degree=3, gamma=0.01, kernel='poly',

max\_iter=-1, probability=False, random\_state=None, shrinking=True,

tol=0.001, verbose=False

Summary

The random forest classifier did a much better job than the support vector machine. It was able to model quicker and with significantly higher accuracy. The SVC is more efficient and should have better performance with two class problems instead of multiclass problems.

Part Two: MNIST Classification Using Keras

Summary

[TODO]